Problem 1 Let:

LAST NAME: May 13 /2013

FIRST NAME: Solution

$$L = \{b^n a^k b^\ell a^j c^m \mid \ell = n,\ j > 2,\ k = 0,\ n, k, \ell, j, m \geq 0\}$$

(a) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer:

The template is $b^{2n} a^{j+3} c^m$, $n, j, m \geq 3$

whence the grammar: $G = (V, \Sigma, P, S)$

$V = \{S, K, B, A\}$, $\Sigma = \{a, b, c\}$,

P: $S \to BAaaaK$

$B \to bbB \mid \lambda$

$A \to aA \mid \lambda$

$K \to cK \mid \lambda$

(b) Write a regular expression that defines L. If such a regular expression does not exist, prove it.

Answer:

$(bb)^* a^* aaa c^*$

Problem 2 Let:

LAST NAME:

FIRST NAME: Solution

$$L = \{c^n b^k c^\ell b^j a^m \mid k = \ell = m,\ j = 0,\ n, k, \ell, j, m \geq 0\}$$

(a) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer: The template is $c^n b^\ell c^\ell a^\ell$, $n, \ell \geq 0$ and the grammar does not exist since L is not context-free. To prove it, assume the opposite. Observe that every string has the property:

(*) $\boxed{\text{\#a's = \#'bs = \#c's after the last b.}}$

Let k be the constant of the Pumping Lemma. Select the word $\overline{w_0 = b^\ell c^\ell a^\ell}$, where ℓ is selected so that $\ell > k$. In any pumping decomposition, the pumping window is shorter than k and shorter than ℓ, hence at least one letter in never pumped and at least one is violating property (*).

(b) Draw a state transition gfraph of a finite automaton that accept L. If such an automaton does not exist, prove it.

Answer:

Impossible, since L is not regular. If L was regular, it would be context-free, since all regular languages are context-free. By the result of part (a), L is not context-free. Hence, L cannot be regular.

Problem 1 Let:

LAST NAME: May 13/2013

FIRST NAME: Solution

$$L = \{b^n a^k b^\ell a^j c^m \mid \ell = n,\ j > 2,\ k = 0,\ n, k, \ell, j, m \geq 0\}$$

(a) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer:

The template is $b^{2n} a^{j+3} c^m$, $n, j, m \geq 0$

whence the grammar: $G = (V, \Sigma, P, S)$

$V = \{S, K, B, A\}$, $\Sigma = \{a, b, c\}$,

P: $S \to BAaaaK$

$B \to bbB \mid \Lambda$

$A \to aA \mid \Lambda$

$K \to cK \mid \Lambda$

(b) Write a regular expression that defines L. If such a regular expression does not exist, prove it.

Answer:

$(bb)^* a^* aaa c^*$

Problem 2 Let:

LAST NAME:

FIRST NAME: Solution

$$L = \{c^n b^k c^\ell b^j a^m \mid k = \ell = m,\ j = 0,\ n, k, \ell, j, m \geq 0\}$$

(a) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer: The template is $c^n b^\ell c^\ell a^\ell$, $n, \ell \geq 0$ and the grammar does not exist since L is not context-free. To prove it, assume the opposite. Observe that every string has the property:

(*) $\boxed{\#a\text{'s} = \#\text{'b's} = \#c\text{'s after the last } b.}$

Let k be the constant of the Pumping Lemma. Select the word $w_0 = b^\ell c^\ell a^\ell$, where ℓ is selected so that $\ell > k$. In any pumping decomposition, the pumping window is shorter than k and shorter than ℓ, hence at least one letter in never pumped and at least one is violating property (*).

(b) Draw a state transition gfraph of a finite automaton that accept L. If such an automaton does not exist, prove it.

Answer:

Impossible, since L is not regular. If L was regular, it would be context-free, since all regular languages are context-free. By the result of part (a), L is not context-free. Hence, L cannot be regular.

Problem 3 Let:

LAST NAME: ____________________

FIRST NAME: Solution

$$L = \{a^n c^k a^\ell c^j b^m \mid j = \ell = n,\ m > 1,\ k = 0,\ n, k, \ell, j, m \geq 0\}$$

(a) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer:

The template is : $a^{2n} c^n bbb^m$, $n, m \geq 0$

whence the grammar : $G = (V, \Sigma, P, S)$

$V = \{S, A, B\}$, $\Sigma = \{a, b, c\}$

P: $S \to AbbB$

$A \to aaAc \mid \Lambda$

$B \to bB \mid \Lambda$

(b) Draw a state transition graph of a finite automaton that accepts L. If such an automaton does not exist, prove it.

Answer: Impossible since L is not regular. Assume the opposite. Observe that every string of L has the property: [# a's = twice the # c's.] (*)
Let k be the constant of the Pumping Lemma. Select a word $w_0 = a^{2n} c^n b^{m+2}$ where n is selected so that $n > k$.
In any pumping decomposition, the pumping window is shorter than k and shorter than n and thus consists of a's only. Pumping up once violates (*).

Problem 4 Let L be the language accepted by the pushdown automaton: $M = (Q, \Sigma, \Gamma, \delta, q, F)$ where: $Q = \{q, p\}$, $\Sigma = \{a, b, c, d, e\}$, $\Gamma = \{A, E, M, X\}$, $F = \{p\}$ and the transition function δ is defined as follows:

$$
\begin{array}{l}
[q, e, \lambda, p, EXAM] \\
[p, a, A, p, \lambda] \\
[p, a, E, p, \lambda] \\
[p, b, M, p, \lambda] \\
[p, c, X, p, \lambda] \\
[p, d, \lambda, p, \lambda]
\end{array}
$$

(Recall that M is defined so as to accept by final state and empty stack. Furthermore, if an arbitrary stack string, say $X_1 \ldots X_n \in \Gamma^*$ where $n \geq 2$, is pushed on the stack by an individual transition, then the leftmost symbol X_1 is pushed first, while the rightmost symbol X_n is pushed last.)

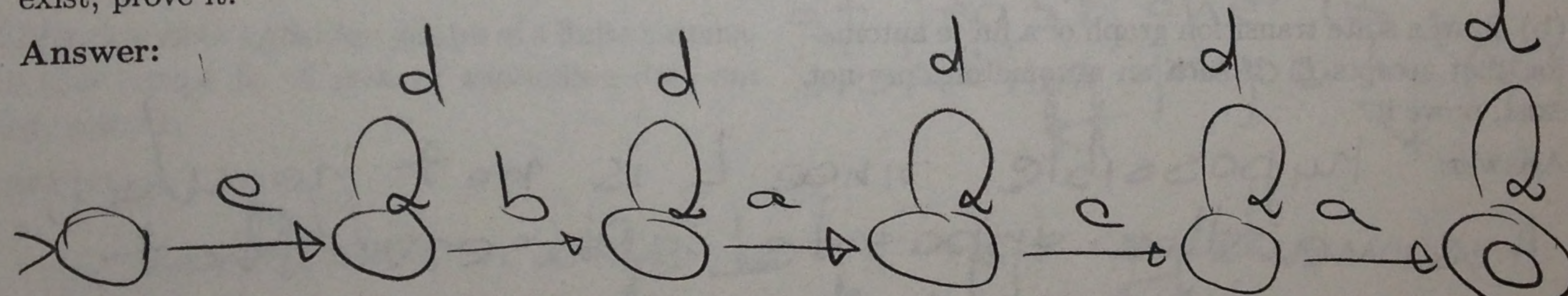
(a) List 6 distinct strings that belong to L. If this is impossible, state it and explain why.

Answer: Advice: L is

$ed^*bd^*ad^*cd^*ad^*$

(b) Draw a state transition graph of a finite automaton that accepts L. If such an automaton does not exist, prove it.

Answer:



LAST NAME:

FIRST NAME: Solution

(c) What is the cardinality of the set L? If it is finite, state the exact number; if it is infinite, state whether it is countable or uncountable.

Answer: L is infinite and countable.

(d) What is the cardinality of the set $\mathcal{P}(L)$ (the set of subsets of L?) If it is finite, state the exact number; if it is infinite, state whether it is countable or uncountable.

Answer: $\mathcal{P}(L)$ is infinite and uncountable.

LAST NAME:

FIRST NAME: Solution

Problem 5 Let L be the language accepted by the pushdown automaton: $M = (Q, \Sigma, \Gamma, \delta, q, F)$ where: $Q = \{q, p\}$, $\Sigma = \{a, b, c, d, e\}$; $\Gamma = \{A, E, M, X\}$, $F = \{p\}$ and the transition function δ is defined as follows:

$$[q, e, \lambda, q, EX]$$
$$[q, e, \lambda, q, AM]$$
$$[q, \lambda, \lambda, p, \lambda]$$
$$[p, b, E, p, \lambda]$$
$$[p, a, X, p, \lambda]$$
$$[p, c, A, p, \lambda]$$
$$[p, d, M, p, \lambda]$$

(Recall that M is defined so as to accept by final state and empty stack. Furthermore, if an arbitrary stack string, say $X_1 \ldots X_n \in \Gamma^*$ where $n \geq 2$, is pushed on the stack by an individual transition, then the leftmost symbol X_1 is pushed first, while the rightmost symbol X_n is pushed last.)

(a) List 6 distinct strings that belong to L. If this is impossible, state it and explain why.

Answer:

λ, eab, edc, eeabdc, eeabab, eedcab

(b) Write a complete formal definition of a context-free grammar that generates L. If such a grammar does not exist, prove it.

Answer:

$G = (V, \Sigma, P, S)$

$V = \{S\}$, $\Sigma = \{a, b, c, d\}$

P:

$S \to eSab \mid eSdc \mid \lambda$

(c) State one trivial property of the language L, such that $\boldsymbol{a^*b^*}$ does not have this property. Explain carefully why this property is trivial, and prove that L indeed has it, while $\boldsymbol{a^*b^*}$ does not. If such a property does not exist, state it, and explain why it is so.

Answer:

Impossible. If this property existed, it would be true for L and false for a^*b^* and by definition could not be trivial (which always assumes the same value.)

(d) State one non-trivial property of the language L, such that $\boldsymbol{a^*b^*}$ does not have this property. Explain carefully why this property is non-trivial, and prove that L indeed has it, while $\boldsymbol{a^*b^*}$ does not. If such a property does not exist, state it, and explain why it is so.

Answer:

Such a property is "every nonempty string begins with e"

L has this property, by its grammar. a^*b^* does not have it, since no string in a^*b^* contains e. Hence the property has different values for two languages, and is by definition non-trivial.

Problem 6 Consider the following Turing machine: $M = (Q, \Sigma, \Gamma, \delta, q, F)$ such that:
$Q = \{q, r, s, p, v, t, z, x, y\}$;
$\Sigma = \{0, 1\}$; $\Gamma = \{B, 0, 1\}$; $F = \{x\}$; and δ is defined by the following transition set:

$[q, 0, r, 0, R]$	$[v, 0, x, 0, L]$
$[r, 1, s, 1, R]$	$[v, 1, z, 1, L]$
$[s, 0, t, 0, R]$	
	$[z, 0, y, 0, L]$
$[t, 0, p, 0, R]$	$[z, 1, x, 1, L]$
$[t, 1, p, 1, R]$	
	$[y, 0, y, 0, R]$
$[p, 0, p, 0, R]$	$[y, 1, y, 1, R]$
$[p, 1, p, 1, R]$	$[y, B, y, B, R]$
$[p, B, v, B, L]$	

(Assume that M is defined so as to have an one-way infinite tape (infinite to the right only.) B is the designated blank symbol. M accepts by final state.)

Let L be the set of strings on which M *diverges*.

(a) List 6 distinct strings that belong to L. If this is impossible, state it and explain why.

Answer: Advice

See part (b).

(b) Write a regular expression that defines L. If such a regular expression does not exist, prove it.

Answer:

$010(0 \cup 1)^* 01 \cup 0101$

LAST NAME:

FIRST NAME:

(c) Dangerous Professor has told her students to write a program that operates as follows:

INPUT: String w over $\{0, 1\}$.

OUTPUT: **yes** if w represents a Turing Machine which halts exactly when the Turing Machine M (defined at the beginning of this problem) diverges;
no otherwise.

Explain the algorithm that should be employed by this program, or state that it does not exist, and prove it.

Answer: Impossible.
If this algorithm existed, it would decide the set of TMs whose languages have the property "accepts by halting the set of strings on which M diverges", or in short, it would decide TMs whose languages have the property "= L". Since L has this property but say ∅ does not, property is non-trivial, and by Rice's theorem, the construction is impossible.

Problem 7 Consider the following Turing machine: $M = (Q, \Sigma, \Gamma, \delta, q)$ such that:
$Q = \{q, p, v, z, x\}$;
$\Sigma = \{0, 1\}$; $\Gamma = \{B, 0, 1, N\}$; $F = \{x\}$; and δ is defined by the following transition set:

$[q, 0, p, N, R]$ $[v, 1, v, 1, L]$
$[q, 1, q, 1, R]$ $[v, 0, x, 0, R]$
$[q, B, q, B, R]$ $[v, N, z, 0, R]$

$[p, 0, p, 0, R]$
$[p, 1, p, 1, R]$
$[p, B, v, B, L]$

(Assume that M is defined so as to have an one-way infinite tape (infinite to the right only.) B is the designated blank symbol. M accepts by final state.)

Let L be the set of string which M *rejects*.

(a) List 6 distinct strings that belong to L. If this is impossible, state it and explain why.

Answer: Advice

See part (b)

(b) Write a regular expression that defines L. If such a regular expression does not exist, prove it.

Answer:

$1^* 0 1^*$

LAST NAME:

FIRST NAME: Solution

(c) Dangerous Professor has told her students to write a program that operates as follows:

INPUT: String w over $\{0, 1\}$.

OUTPUT: **yes** if w represents a Turing Machine that accepts exactly those strings which the Turing Machine M (defined at the beginning of this problem) rejects;
no otherwise.

Explain the algorithm that should be employed by this program, or state that it does not exist and prove it.

Answer:

Impossible. If this algorithm existed it would decide the set of TMs whose languages have the nontrivial property "is equal to L".

Since the property is true for L but false say for $\emptyset$, it is nontrivial and by Rice's theorem, the construction is impossible.

Problem 8 Consider the following Turing machine: $M = (Q, \Sigma, \Gamma, \delta, q)$ such that:
$Q = \{q, p, v, z, x\}$;
$\Sigma = \{0,1\}$; $\Gamma = \{B, 0, 1\}$; $F = \{x\}$; and δ is defined by the following transition set:

$[q, 0, q, 0, R]$ $[v, 0, x, 0, R]$
$[q, 1, p, 1, R]$ $[v, 1, z, 1, R]$
$[q, B, q, B, R]$

$[p, 1, q, 1, R]$
$[p, 0, p, 0, R]$
$[p, B, v, B, L]$

(Assume that M is defined so as to have an one-way infinite tape (infinite to the right only.) B is the designated blank symbol. M accepts by final state.)

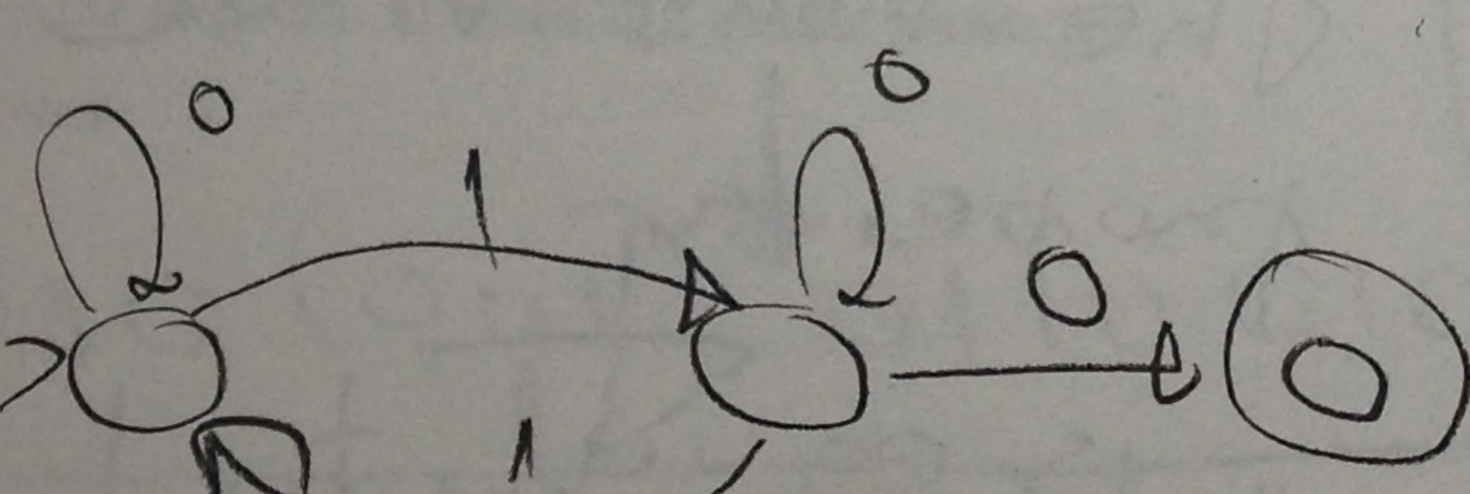
Let L be the set of string which M *accepts*.

(a) List 6 distinct strings that belong to L. If this is impossible, state it and explain why.

Answer: Advice: contains an odd # 1's and ends with 0.

(b) Draw a state transition graph of a finite automaton that accepts L. If such an automaton does not exist, prove it.

Answer:



LAST NAME:

FIRST NAME: Solution

(c) Dangerous Professor has told her students to write a program that operates as follows:

INPUT: String w over $\{0, 1\}$.

OUTPUT: **yes** if w is a string such that the Turing Machine M (defined at the beginning of this problem) accepts w;
no otherwise.

Explain the algorithm that should be employed by this program, or state that it does not exist and prove it.

Answer:

Convert the finite automaton obtained in the answer to part (b) to a deterministic equivalent, simulate this deterministic automaton, and decide exactly as it does.

LAST NAME: Solution

FIRST NAME:

Problem 9 Let L be the set of all strings over the alphabet $\{a, b, c\}$ which satisfy all of the following properties.

- if the string begins with a, then it contains an odd number of a's.
- if the string begins with b, then all of the following conditions hold:
  1. the string ends with b;
  2. the string has an odd length;
  3. the middle symbol is equal to the last symbol;
- if the the string begins with c, then both of the following conditions hold:
  1. the string has an even length;
  2. the string is a palindrome;

Write a complete formal definition of a context-free grammar that generates the language L. If such a grammar does not exist, state it and explain why.

Answer:

$G = (V, \Sigma, P, S)$

$\Sigma = \{a, b, c\}$, $V = \{S, A, D, E, B, M, Z, K, L\}$

P: $S \to A \mid B \mid K$

$A \to aD$

$D \to aE \mid bD \mid cD \mid \lambda$

$E \to aD \mid bE \mid cE$

$B \to bMb \mid b$

$M \to ZMZ \mid b$

$Z \to a \mid b \mid c$

$K \to cLc$

$L \to aLa \mid bLb \mid cLc \mid \lambda$